Data Augmentation using Clustering Based Techniques

V S N Lokesh Yarramallu - AM.EN.U4AIE22055

Sri Kaushik Kesanapalli AM.EN.U4AIE22026

Rahul Nallagattu AM.EN.U4AIE22036

Koti Reddy Ambati AM.EN.U4AIE22004

Chapter-1: Introduction

Data Augmentation is a pivotal technique in machine learning, which creates modified or synthetic data points to expand the size and diversity of datasets. The goal of this work is to improve numerical datasets by the deployment of cluster-based strategies for data augmentation. Utilizing clustering techniques, we find organic groupings in the data and create new synthetic data points while maintaining the original data's inherent patterns and statistical characteristics. Using scaling, noise injection, and feature augmentation, our method ensures the enhanced data increases the dataset's complexity and coverage. In addition to offering a richer training set that enhances model performance, this approach turns out to be more time-efficient than conventional data collection.

1.1 Motivation

The motivation behind this study stems from the need to address several key challenges in the field of machine learning, particularly concerning the limitations of numerical datasets

- **Dataset Expansion**: Data augmentation allows for the creation of additional data points, thereby expanding the size of the dataset beyond its original scope.
- **Enhanced Coverage**: By generating synthetic data points that span different regions of the feature space, data augmentation improves the coverage of the dataset, ensuring a more comprehensive representation of the underlying data distribution.
- **Improved Learning**: The increased diversity and quantity of data facilitate better learning by machine learning models. This variety helps models generalize better to unseen data, reducing overfitting and improving overall performance.

- **Supports Model Complexity**: Augmenting the dataset with diverse examples supports the training of more complex models that can capture intricate patterns and relationships in the data.
- **Time Efficient Data Collection**: Instead of gathering new data through costly and time-consuming methods, data augmentation provides a more efficient way to increase dataset size and diversity, accelerating the model development and deployment process.

1.2 Problem Statement

A basic machine learning approach called "data augmentation" is used to create modified or synthetic versions of existing data points to expand the size and diversity of a dataset. In situations when gathering a significant amount of labelled data is difficult or resource-intensive, this procedure is crucial. Data augmentation contributes to the improvement of machine learning models' robustness and generalizability by enriching the dataset. The usage of data augmentation has primarily concentrated on text and image data in recent years. Applying these methods to numerical datasets is becoming increasingly popular, nevertheless. Our method focuses on numerical data, using cluster-based methods to provide augmented data while maintaining the underlying patterns and features of the original dataset. In cluster-based data augmentation, new data points are created inside existing clusters by organizing data points into clusters according to their properties. By using this method, the distribution and statistical characteristics of the original data are preserved in the enhanced data. Synthetic data points are created using methods including scaling, noise injection, and feature augmentation to improve the dataset and model performance. This study examines the use of cluster-based methodologies for data augmentation, emphasizing the advantages, procedures, and useful uses. The goal is to show how these methods may be applied to enhance the number and quality of numerical datasets, which would enable machine learning models that are more dependable and accurate.

1.3 Objectives

The objectives of this project are

- **Implement Cluster-Based Data Augmentation**: Develop and implement methodologies to augment existing datasets using clustering techniques such as K-means and hierarchical clustering.
- **Generate Synthetic Data Points**: Utilize clustering results to generate synthetic data points that reflect the characteristics and distributions observed in the original dataset.
- **Preserve Data Patterns**: Ensure that the generated data points preserve the statistical properties and patterns present in the original dataset.

- **Enhance Dataset Size and Diversity**: Increase the size and diversity of the dataset by incorporating augmented data points derived from clustering methods.
- **Support Machine Learning Model Training**: Facilitate the training of more robust machine learning models by providing a larger and more varied dataset, thereby improving model generalization and performance.
- **Evaluate Impact of Augmented Data**: Assess the effectiveness of the augmented dataset in improving model accuracy and robustness compared to using the original dataset alone.

1.4 Contribution

The cluster-based approaches for data augmentation project adds a great deal to the field of machine learning. It creates a methodological foundation for using clustering algorithms like hierarchical clustering and K-means to enhance datasets. The research successfully creates synthetic data points by applying these techniques, which resemble the statistical distributions and patterns present in the original datasets. By using this method, the datasets become larger and more diverse, which is important for enhancing the resilience and generalizability of machine learning models. Furthermore, the project guarantees the quality and integrity of the augmented datasets by maintaining data patterns throughout augmentation, which helps to promote correct machine learning model training and evaluation. By means of stringent assessment and verification procedures, which encompass parallels with models trained on source datasets, the project highlights the effectiveness of its methodology in enhancing model performance. The project's results also open new avenues for investigation into more sophisticated clustering methods and their wider uses in data augmentation, indicating that machine learning and data science will continue to progress.

Chapter-2:

Data Pre-Processing

2.1 Preamble

A crucial step in the machine learning process, data preprocessing guarantees the dependability, quality, and usability of datasets used for assessment and training. The methods and strategies used to convert unprocessed data into a format that machine learning algorithms can understand are the main topics of this chapter. It covers several preparation procedures, including feature engineering to extract relevant information, feature scaling to normalize the range of features, and data cleaning to manage missing values and outliers. This chapter also looks at how encoding methods can be used to categorical variables to make them compatible with machine learning models that need numerical inputs. Through the explanation of these preprocessing techniques, Chapter 2 lays the groundwork for future chapters' successful model building and evaluation by offering a fundamental understanding of how data quality and integrity are preserved throughout the machine learning workflow.

2.2 Dataset

Landmines Dataset:

The dataset was originally created for research in passive mine detection, utilizing sensor data to distinguish between diverse types of landmines buried underground. It poses challenges due to its sparse and noisy sensor readings, making it suitable for data augmentation techniques aimed at improving classification accuracy.

- **Features**: Includes sensor output voltage (V), sensor height (H), and soil type (S).
- **Type**: Classification
- **Source**: Originally created for passive mine detection, sourced from UCI datasets.
- **Purpose**: Used to classify several types of landmines based on sensor readings and soil type.

Chronic Kidney Disease Dataset:

This dataset, which was gathered over an extended period in India, includes a wide range of health markers that are essential for identifying chronic kidney disease. It has a lot of numerical and categorical information, and treating missing data and standardization carefully is typically necessary before using data augmentation techniques.

- **Features**: Contains 24 features including age, blood pressure, urine characteristics (e.g., presence of red blood cells), and biochemical parameters (e.g., blood glucose levels).
- **Type**: Classification
- **Source**: Collected over a period in India, sourced from UCI datasets.
- **Purpose**: Used to classify patients into chronic kidney disease (CKD) or non-CKD categories based on health indicators.

Estimation of Obesity Dataset:

This dataset sheds light on the variables affecting obesity rates in various demographic groups. It provides sophisticated analysis and classification jobs using features pertaining to physical activity, food, and lifestyle choices. By increasing the diversity of the dataset, data augmentation approaches can strengthen the predictive power of models for obesity levels.

- **Features**: Includes 16 features related to dietary habits, physical activity levels, and lifestyle choices across multiple countries.
- **Type**: Classification
- **Source**: Sourced from UCI datasets.
- **Purpose**: Used to estimate and classify obesity levels based on lifestyle and health-related factors.

Rice Type Classification Dataset:

This dataset, which was created for instructional purposes, provides a simple categorization job based on the physical characteristics of rice grains. To improve classification accuracy, this dataset can be enhanced by creating artificial data points to imitate fluctuations in rice grain attributes or to balance class distributions.

- **Features**: Consists of 7 features such as area, perimeter, and shape descriptors of rice grains.
- **Type**: Classification
- **Source**: Developed for educational and research purposes, sourced from UCI datasets.

- **Purpose**: Used to classify several types of rice based on physical characteristics of rice grains.

2.3 Implementation

2.3.1 Data Preprocessing Techniques

Data preprocessing is a crucial step in preparing raw data for machine learning models. It involves several techniques to clean, transform, and prepare data to ensure its quality and suitability for analysis.

Imputing:

Imputing refers to the process of filling in missing values in a dataset. Missing data can occur due to distinct reasons such as data collection errors or incomplete records.

- **Simple Imputation**: This method fills missing values with a central tendency measure of the data, such as mean, median, or mode. It is straightforward and effective for handling missing data in numeric and categorical features.
- **Random Value Imputation**: In this approach, missing values are replaced with randomly selected values from the same column. This technique can introduce variability into the dataset and is useful when missing data is not systematic.

Encoding:

- **Label Encoding**: Converts categorical variables into numerical labels. Each category is assigned a unique integer. It is suitable for ordinal data where there is a meaningful order among categories but not for nominal data where categories are unordered.

Outliers:

- **Box Plot Outliers**: Outliers are data points that significantly differ from other observations in the dataset. Box plots visually display the distribution of data and identify outliers based on statistical methods like the interquartile range (IQR). Outliers can be treated by removing them, transforming them, or using robust algorithms that are less sensitive to outliers.

Standard Scaling:

Also known as Z-score normalization, standard scaling transforms data to have a mean of 0 and a standard deviation of 1. It is particularly useful for algorithms that assume normally distributed data or those that require features to be on the same scale, such as linear regression, logistic regression, and K-means clustering.

Data preprocessing ensures that the data is clean, consistent, and ready for analysis. These techniques help in handling missing values, converting categorical data into numerical formats, addressing outliers, and standardizing data to improve the performance and accuracy of machine learning models.

Principal Component Analysis (PCA):

PCA is a data preprocessing technique used for dimensionality reduction. It simplifies a dataset by transforming it into a smaller set of uncorrelated variables called principal components, preserving the most valuable information. Some functionalities are:

- Dimensionality Reduction: Simplifies the dataset by capturing the most variance with fewer components.
- Feature Extraction: Creates new variables as linear combinations of original features, ordered by variance.
- Noise Reduction: Focuses on components with the most variance, reducing noise and improving data quality.
- Visualization: Projects high-dimensional data onto a lower-dimensional space for easier visualization.
- Improving Model Performance: Prevents overfitting by reducing feature count, enhancing model robustness.

PCA is used for tasks like image compression, noise filtering, and feature reduction, making the dataset simpler and more suitable for analysis.

2.3.2 Detailed Application of Preprocessing Techniques

1. Landmines Dataset

The TSI dataset was initially loaded and inspected to ensure data integrity and suitability for analysis. It includes features such as voltage, height, soil type, and my type. The dataset was thoroughly described to understand its structure and statistical properties, ensuring comprehensive insights into the dataset's characteristics. The data is cleaned and normalized initially.

Outliers within the dataset were identified using the Interquartile Range (IQR) method. Instead of removing outliers, which could lead to information loss, they were retained to preserve the dataset's full variability and maintain real-world fidelity in subsequent analyses to prepare the data for modelling, numerical features were standardized using StandardScaler. This step was crucial for ensuring fair contribution from all features, especially beneficial for algorithms like KMeans clustering that rely on distance metrics. Overall, these preprocessing steps were designed to enhance the dataset's robustness, facilitate accurate analysis, and preserve the integrity of its statistical properties for effective machine learning model development and evaluation.

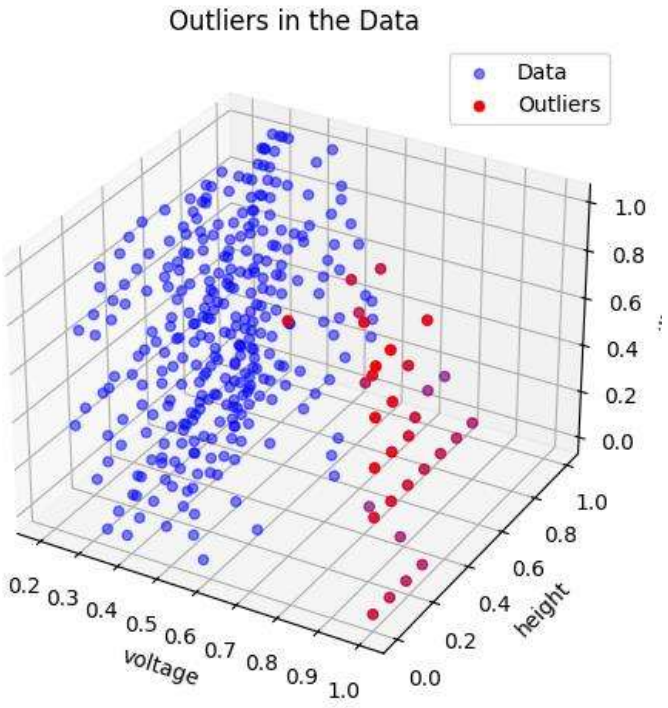


Figure 1: Outliers in Landmines Dataset

2. Chronic Kidney Disease Dataset

The dataset underwent meticulous preprocessing to ensure its integrity and suitability for analysis. Initially, significant efforts were directed towards rectifying typographical errors and formatting inconsistencies across columns, ensuring uniformity and clarity in data representation. Missing data were handled using appropriate imputation strategies tailored to each feature's characteristics: numerical columns were imputed using random sampling, while categorical columns were filled with mode values to maintain data integrity. Subsequently, the dataset underwent rigorous cleaning and normalization procedures to standardize feature scales and ensure robustness against outliers.

Outliers were identified and filtered using the interquartile range (IQR) method, maintaining statistical integrity while enhancing data quality. Feature encoding was essential, employing Label Encoder to convert categorical variables into numerical formats suitable for machine learning algorithms, facilitating comprehensive data analysis. Moreover, advanced techniques for feature reduction were systematically applied for better performance and feature capturing, a total of top 10 features are selected. Clustering and will be elaborately discussed in Chapter 3, aiming to streamline model complexity, enhance interpretability, and optimize computational efficiency.

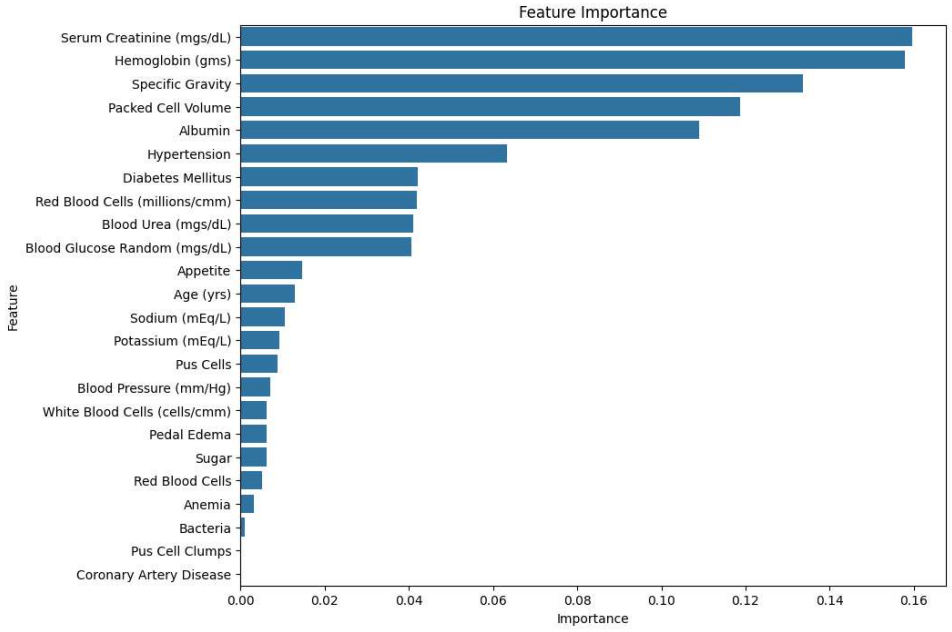


Figure 2: Feature Importance of Kidney Disease Dataset

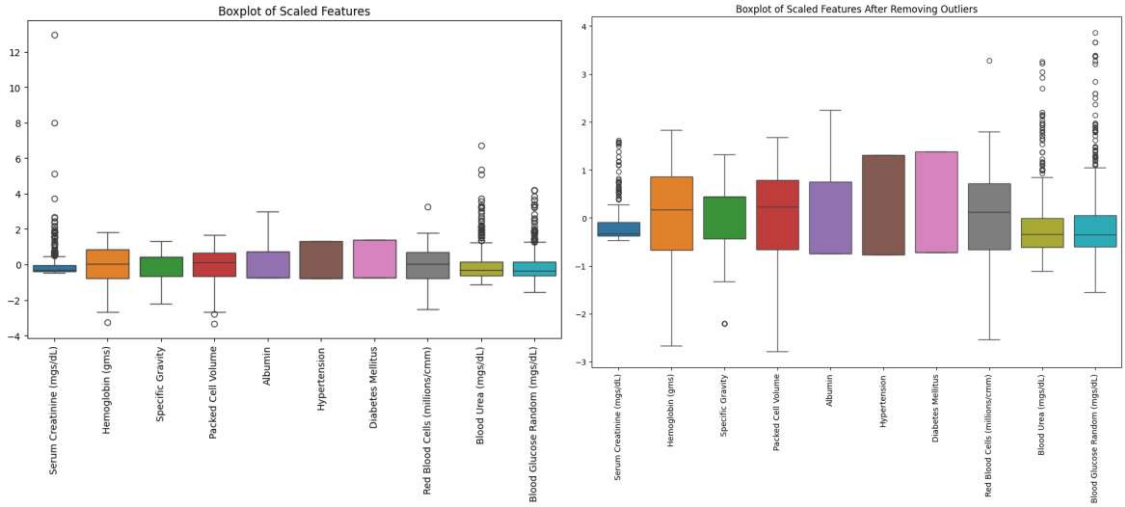


Figure 3: Boxplot before and after removing Outliers

3. Obesity Dataset

The Obesity dataset undergoes preprocessing to ensure it meets the requirements for detailed analysis. Initially, the dataset is imported into a Pandas Data Frame for comprehensive exploration. Key steps include handling missing values, where strategies such as imputation or removal are applied to maintain data integrity. Categorical variables are encoded using a label encoding technique to transform them into numerical format suitable for machine learning algorithms. Numerical features are scaled using standardization techniques to normalize their values across the dataset. Exploratory data analysis (EDA) techniques, such as statistical summaries and visualizations like box plots, provide insights into the distribution and characteristics of the dataset.

Feature selection methods are considered to identify the most relevant features for modeling and selected 7 features, while dimensionality reduction techniques like Principal Component Analysis (PCA) are planned for implementation in chapter 3. This rigorous preprocessing ensures the Obesity dataset is prepared for subsequent stages of analysis, emphasizing robust data preparation to yield reliable insights into obesity-related factors and predictions.

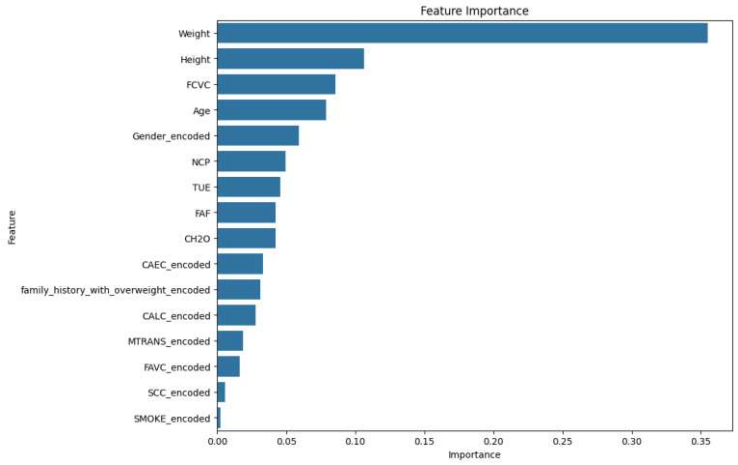


Figure 4: Feature importance of Obesity Dataset

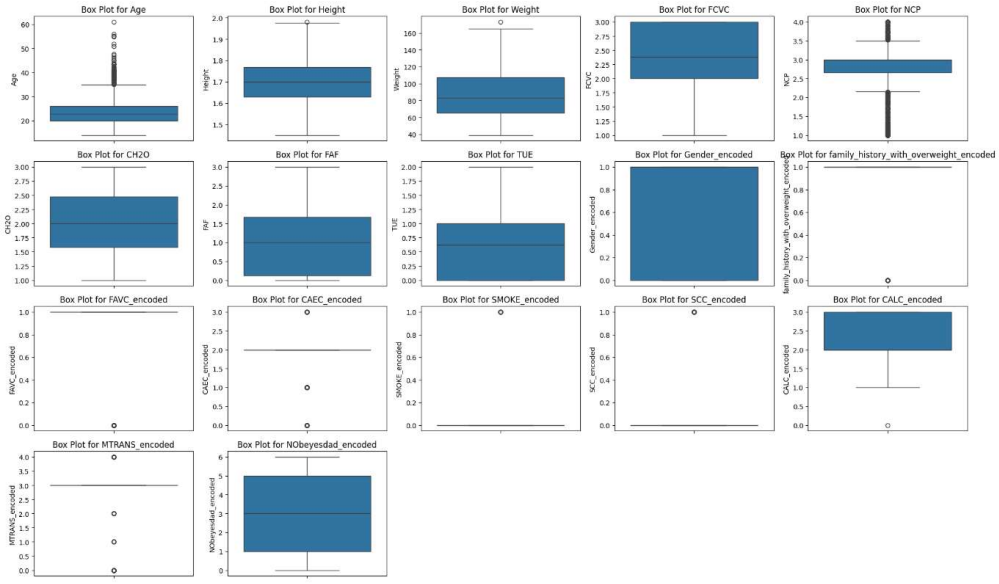


Figure 5: Box plot of each feature of the dataset

4. Rice Classification

Rice classification is done based on physical attributes, we meticulously preprocessed the Rice Classification dataset to ensure its suitability for analysis and model training. Initially, we loaded the dataset and conducted exploratory data analysis to understand its structure, identifying any missing values. To encode the target variable, which denotes rice varieties, we applied label encoding using Label Encoder from scikit-learn. Handling missing

values was accomplished through mean imputation using Simple Imputer, ensuring data completeness without compromising integrity.

Subsequently, we standardized the numerical features using StandardScaler to maintain uniformity and enhance model performance in subsequent steps. Visual inspection through boxplots confirmed the effectiveness of scaling, demonstrating consistent data distribution across features. This meticulous preprocessing prepares the Rice Classification dataset for robust analysis and accurate modeling, setting the stage for insightful conclusions regarding rice variety classification based on physical attributes. We selected the top 5 features from the Boxplot and Coefficient matrix.

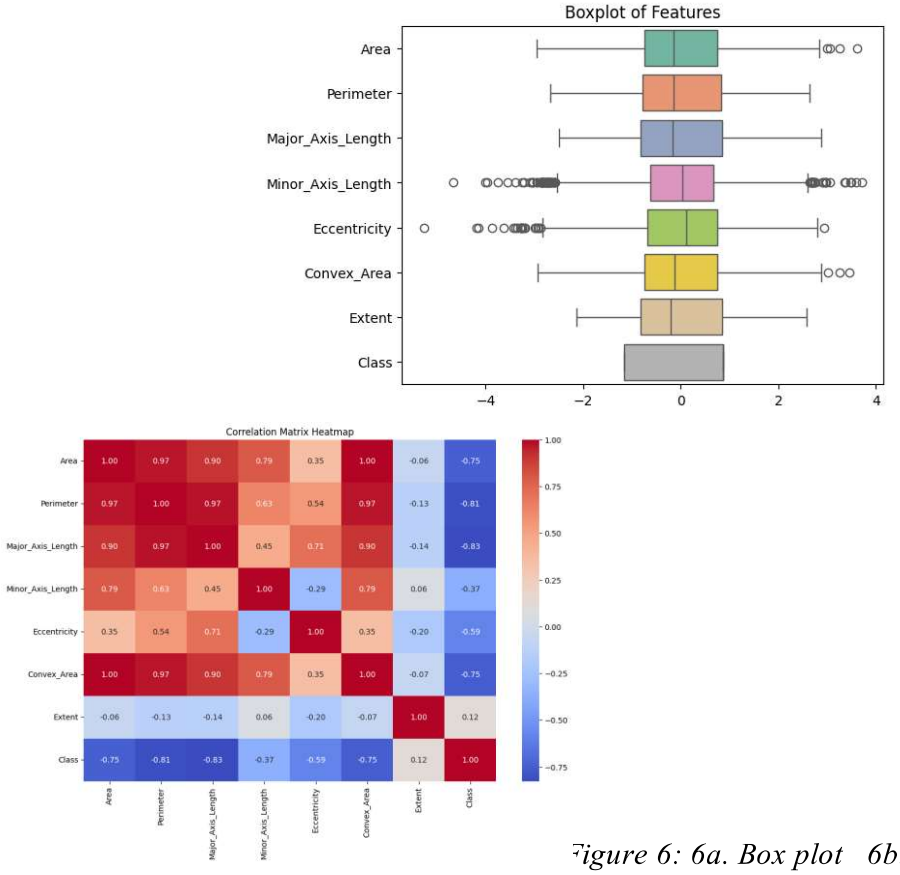


Figure 6: 6a. Box plot 6b. Heat Map of all the features

2.4 Discussion

In this section, we discuss the preprocessing techniques applied to multiple datasets—Landmines, Chronic Kidney Disease, Obesity, and Rice Classification—to ensure data quality, integrity, and suitability for analysis and modeling.

2.4.1 Combined Insights across Datasets

Across all datasets, common preprocessing themes emerged, emphasizing the importance of:

1. **Data Cleaning and Integrity:** Addressing missing data, handling outliers, and ensuring data consistency were foundational steps across all datasets. Techniques such as imputation and outlier management (using methods like IQR) were crucial to maintain dataset integrity.
2. **Standardization and Scaling:** Standardizing numerical features using techniques like StandardScaler was pivotal to ensure fair contribution from all features, facilitating robust model training and evaluation.
3. **Feature Engineering and Selection:** Techniques such as feature encoding (LabelEncoder), feature selection (e.g., SelectKBest, RFE), and planned dimensionality reduction (e.g., PCA) were instrumental in identifying informative features, optimizing model performance, and enhancing interpretability.
4. **Domain-Specific Considerations:** Each dataset presented unique challenges and considerations based on its domain (e.g., medical data for Chronic Kidney Disease and Obesity, physical attributes for Rice Classification). Tailored preprocessing approaches were essential to address specific data characteristics and modeling requirements.

Effective preprocessing lays a solid foundation for successful machine learning applications, influencing model performance, interpretability, and the reliability of insights derived from data analysis. By applying rigorous preprocessing techniques tailored to each dataset's characteristics, the studies discussed here were able to prepare robust datasets for detailed analysis and modeling. Future work will continue to explore advanced modeling techniques and optimization strategies, building upon the preprocessing frameworks established across these diverse datasets to uncover meaningful patterns and support informed decision-making in various domains.

Chapter-3: Clustering and Augmentation

3.1 Introduction

Clustering is a key technique in data analysis and machine learning, grouping objects into clusters based on similarity. This helps in uncovering inherent structures in data, facilitating tasks like data summarization, pattern recognition, and anomaly detection. Clustering simplifies complex datasets, making it easier to identify and interpret patterns. Popular algorithms include K-means, which partitions data into a predefined number of clusters, and hierarchical clustering, which creates a tree-like structure of nested clusters. Density-based methods, like DBSCAN, find clusters based on data density, useful for discovering clusters of arbitrary shapes. Applications range from market segmentation to gene expression analysis, providing crucial insights for decision-making.

Data Augmentation is another vital technique in data analysis and machine learning that increases the diversity of training data by applying transformations to the original data. This enhances the model's ability to generalize and prevents overfitting. For images, common transformations include rotations, translations, and scaling, while text data may undergo noise addition and sequence alterations.

The goal is to expose the model to a wider range of variations, ensuring it learns robust features not specific to the training set. Techniques like flipping, cropping, and color jittering for images, and synonym replacement for text, are widely used. Advanced methods, such as Generative Adversarial Networks (GANs), can also create synthetic data samples.

Data augmentation is essential in fields like computer vision, natural language processing, and speech recognition, improving model performance, especially with limited or imbalanced training data.

3.2 Implementation

3.2.1 Clustering Techniques

There are various techniques for clustering such as K-Means Clustering, DBSCAN (Density-Based Spatial Clustering of Applications with Noise), Hierarchical Clustering, etc... Which can be used based on the data and the goals of our analysis.

K-Means Clustering:

It is one of the most popular and straightforward clustering algorithms. It partitions the dataset into a predefined number of clusters (denoted by 'K'). The algorithm follows these steps:

1. Initialize K centroid randomly.
2. Assign each data point to the nearest centroid, forming K clusters.
3. Update the centroids as the meaning of all points in each cluster.
4. Repeat the assignment and update steps until convergence (i.e., the centroid no longer changes significantly)

   K-means is efficient for large datasets but requires specifying the number of clusters beforehand and can struggle with clusters of varying sizes and densities. For the determination of K, we use The Elbow Method. The Elbow Method helps determine the optimal number of clusters in K-means clustering by identifying the point where adding more clusters does not significantly improve the clustering.

Steps for using Elbow method:

1. **Run K-means for various K values:** Apply K-means with different cluster numbers (e.g., 1 to 10).
2. **Calculate WCSS:** Compute the Within-Cluster Sum of Squares (WCSS) for each K, measuring the compactness of clusters.
3. **Plot WCSS vs. K:** Create a plot with the number of clusters on the x-axis and WCSS on the y-axis.

4. **Identify the Elbow Point:** Look for the point where the WCSS curves bends and flattens, indicating diminishing returns from adding more clusters.

Interpretation:

1. Before the Elbow: WCSS decreases significantly with more clusters.
2.At the Elbow: Optimal balance between cluster count and compactness.
3.After the Elbow: Minimal improvement in WCSS, risking overfitting.

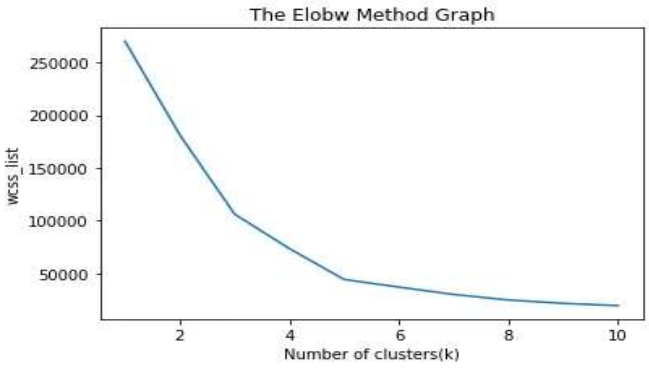


Figure 7: Example of Elbow Method plot

The Elbow Method provides a visual way to choose an appropriate number of clusters.

DBSCAN (Density-Based Spatial Clustering of Applications with Noise):

DBSCAN is a density-based clustering algorithm that identifies clusters based on the density of data points. It operates as follows:

1. Select an arbitrary point and retrieve its neighborhood within a given radius (ε).
2. If the neighborhood contains at least a minimum number of points (MinPts), a new cluster is started.
3. Expand the cluster by including all points within ε of any point in the cluster.
4. Repeat the process for remaining points not yet assigned to a cluster or marked as noise

DBSCAN is effective at finding clusters of arbitrary shapes and handling noise but can be sensitive to the choice of ε and MinPts parameters.

Hierarchical Clustering:

Hierarchical clustering creates a tree-like structure of nested clusters, either by:

- **Agglomerative Approach:** Start with each data point as its own cluster and iteratively merge the closest pairs of clusters until a single cluster remains.
- **Divisive Approach:** Start with the entire dataset as one cluster and iteratively split it into smaller clusters.

The results are typically represented as a dendrogram, which can be cut at distinct levels to form clusters of varying granularities. Hierarchical clustering does not require specifying the number of clusters beforehand and can capture a rich hierarchy of clusters, but it is computationally intensive for large datasets.

3.2.2 Grouping Datasets using clustering techniques

1. Landmines Dataset:

- After applying the Elbow Method, we considered 4 as an optimal value for the number of clusters for effective clustering.
- We applied K-Means Clustering with K = 4.

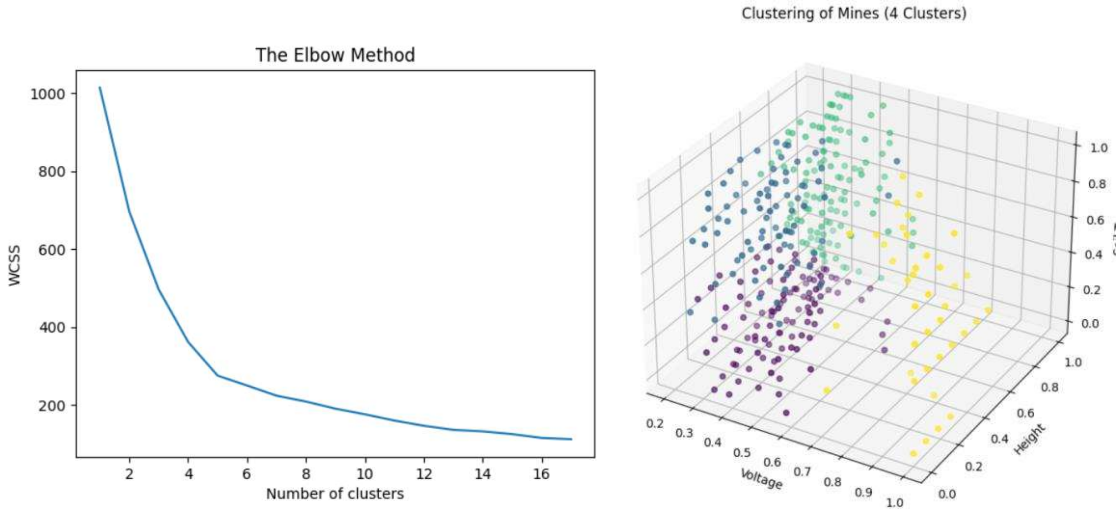


Figure 8: 8a. Elbow method graph for Landmines | 8b. Clusters formed

2. Chronic Kidney Disease Dataset:

- After applying the Elbow method, we considered 4 as an optimal value for the number of clusters for effective clustering.
- We applied K-Means Clustering with K = 4.
- We also use Agglomerative and DBSCAN clustering methods.

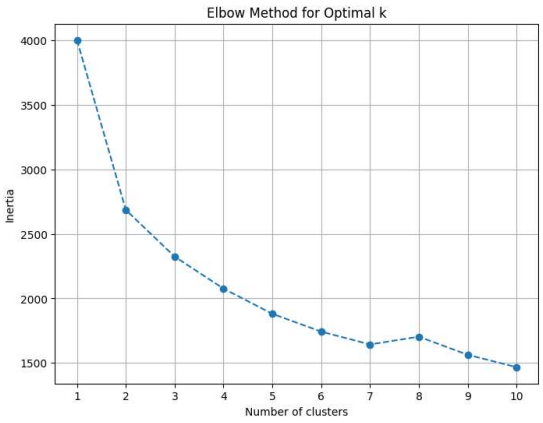


Figure 9: Elbow method plot for Chronic Kidney Disease Dataset

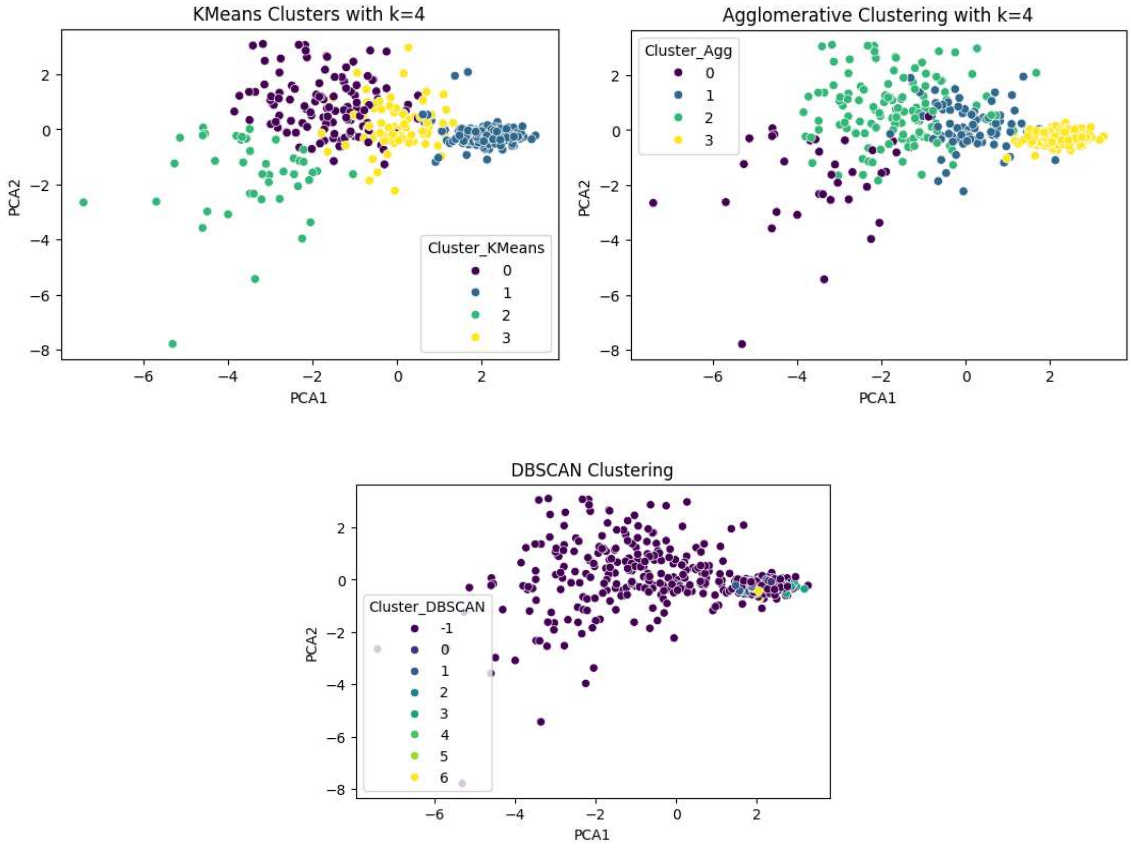


Figure 10: Plot of 2 PCA components in different methods of clustering

3. Obesity Dataset:

- After applying the Elbow method, we considered 4 as an optimal value for the number of clusters for effective clustering.
- We applied K-Means Clustering with K = 4.
- We also used Agglomerative and DBSCAN Clustering methods.

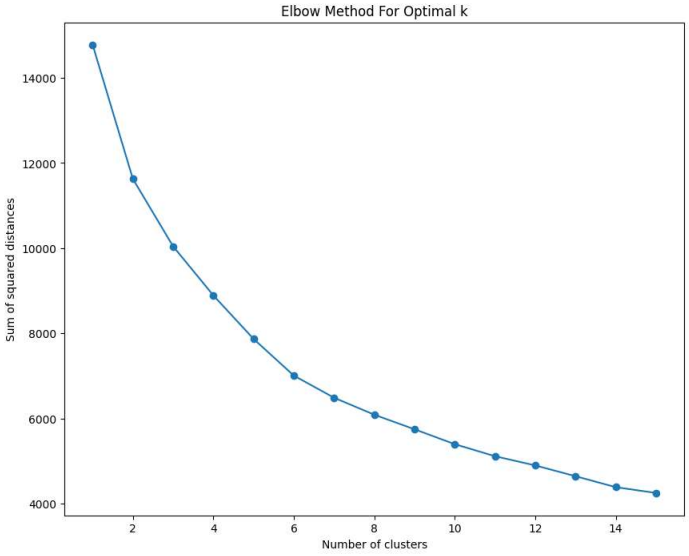


Figure 11: Elbow Method plot for Obesity Dataset

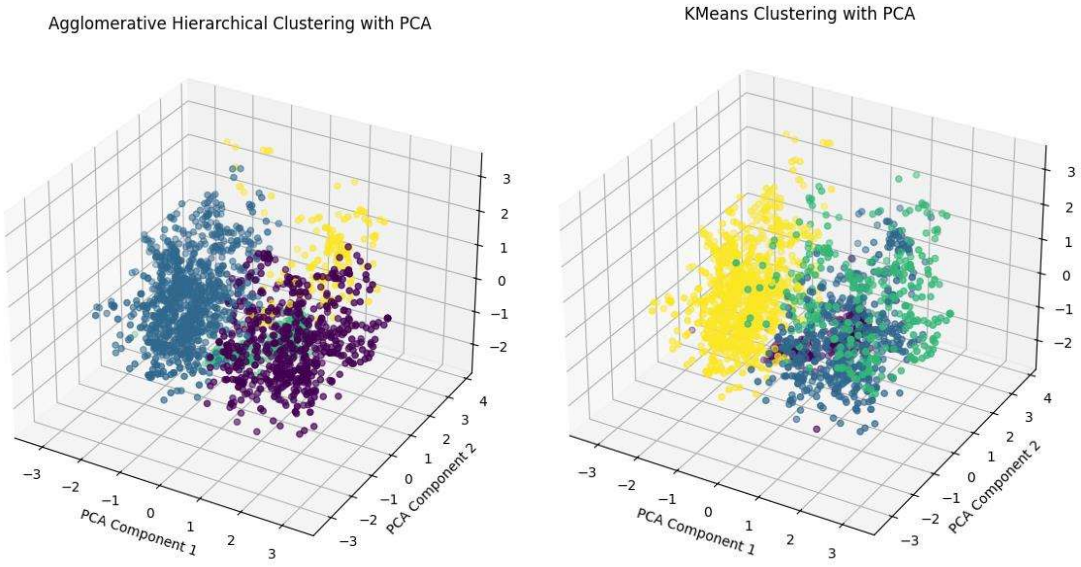


Figure 12: Clustering plot of 3 PCA components in Agglomerative and KMeans Clustering techniques

4. Rice Classification Dataset

The Rice Classification dataset was meticulously preprocessed to prepare for classification based on physical attributes. Initial steps included data loading, imputation of missing values, and encoding of categorical variables using LabelEncoder. Standardization of numerical features with StandardScaler enhanced model performance by ensuring consistent data scales across different attributes. Feature selection techniques, including SelectKBest and Recursive Feature Elimination (RFE), were employed to identify the most relevant features for classification tasks, optimizing model efficiency and interpretability.

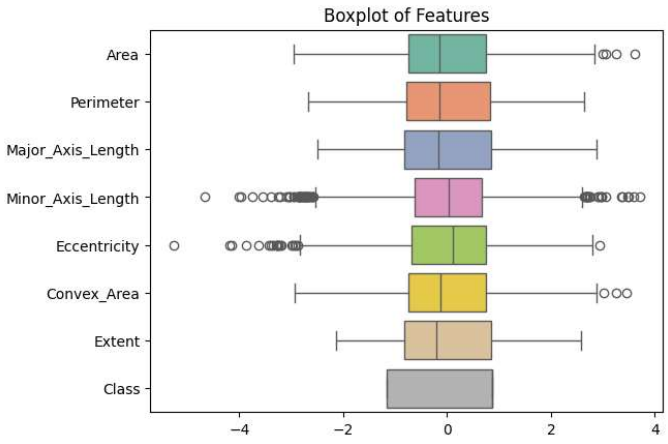


Figure 13: Box Plot of all features of Rice Classification Dataset

3.2.3 Data Augmentation

Data augmentation is a critical technique in machine learning that enhances the diversity and quantity of training data without the need for additional data collection. This report outlines the methodology used to generate synthetic data points for each cluster within an existing dataset. The approach aims to improve the robustness and generalization of machine learning models by introducing controlled variability while maintaining the integrity of the original data.

The primary objective of the data augmentation methodology is:

1. **Enhance Data Diversity:** Introduce variability into the dataset to improve model robustness.
2. **Prevent Overfitting:** Increase the number of training examples, especially in underrepresented clusters, to reduce the risk of overfitting.
3. **Maintain Data Integrity:** Ensure that synthetic data points are realistic and adhere to the statistical properties of the original data.

The methodology involves several key steps:

1. **Define Parameters:**
   - **Number of Data Points:** Specify the number of synthetic data points to generate per cluster.
   - **Noise Levels:** Determine the standard deviation of the noise to be added to each feature, either uniformly or individually per feature.
   - **Feature Types:** Identify features that should be treated as integers or have specific decimal precision.

- **Cluster Identification:** Identify the column in the dataset that includes cluster labels.

2. **Calculate Cluster Means:**
   - Group the date by cluster labels.
   - Compute the mean of each feature within each cluster, resulting in a set of centroid values that represent the central tendency of each cluster.
3. **Generate Synthetic Data:**
   - For each cluster:
     - Create an empty structure to store the synthetic data points.
     - For each feature in the cluster:
       - Generate random noise with a specified standard deviation.
       - Add this noise to the cluster's mean value for the feature to introduce variability.
       - Ensure that the resulting values remain within plausible limits, defined by a percentage range around the feature's original data range.
       - Adjust the values for features that need to be integers or have specific decimal precision.
4. **Adjust and Clip Values:**
   - Apply rounding to features that should be integers.
   - Round decimal features to the specified number of decimal places.
   - Clip all values to a plausible range, ensuring they remain realistic and within the expected bounds of the original data.
5. **Combine Augmented Data:**
   - Append the synthetic data points for each cluster into a single dataset.
   - Ensure that the synthetic data retains the original structure and type consistency of the input data.

6. **Output the Augmented Dataset:**
   - Return the combined dataset containing the original and synthetic data points, ready for further use in training machine learning models.

Practical Considerations:

- **Noise Level Calibration:** It is crucial to choose appropriate noise levels to balance between introducing useful variability and maintaining data realism.
- **Data Range Management:** Augmented values should remain within sensible bounds to avoid unrealistic data points.
- **Feature Type Handling:** Integer and decimal features should be carefully managed to preserve their integrity in the augmented dataset.

This methodology provides several benefits, particularly in scenarios where the original dataset in small or imbalanced:

1. **Enhanced Data Diversity:** By introducing controlled variability, the augmented data helps improve the robustness of machine learning models.
2. **Improved Generalization:** With more training examples, models are better equipped to generalize to unseen data, reducing the risk of overfitting.
3. **Realistic Augmentation:** Ensuring that synthetic data points are realistic and adhere to the original data's statistical properties maintains data integrity and usefulness.

3.2.4 Augmented Data Generation on the Current Dataset

1. Landmines Dataset:

- Original Data is plotted in 4 clusters from the original dataset.
- New Data is also plotted in 4 clusters by newly generating 100 new data points to the original data.

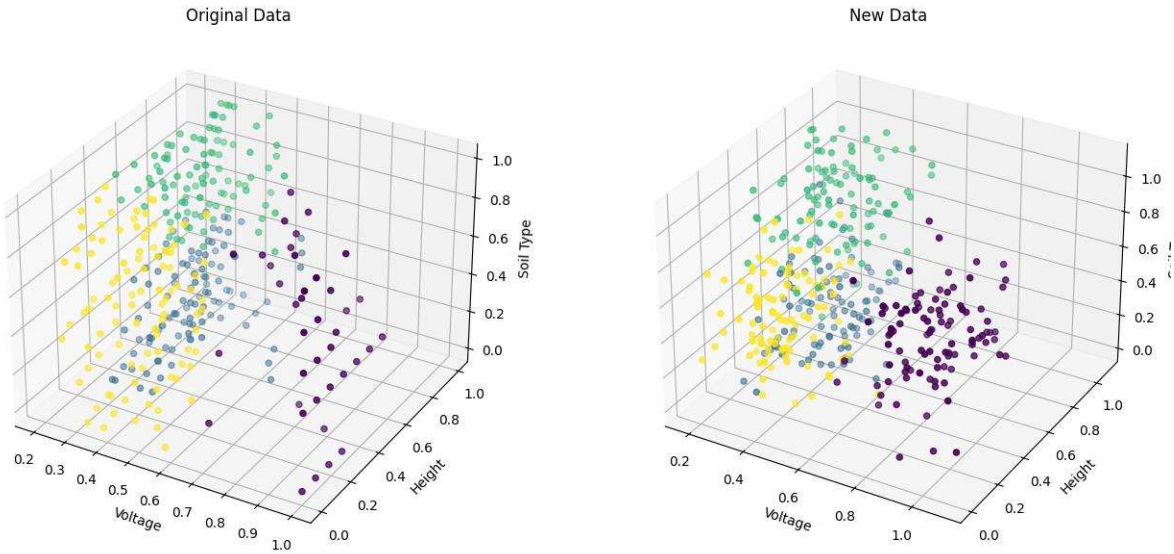


Figure 14: Clustering plot of Original Data and Combined Data using K-Means Clustering

2. Chronic Kidney Disease Dataset:

- After generating 100 new data points from the original dataset, we plotted distribution comparison for 25 features in the dataset to see how the augmented data and original data look like.

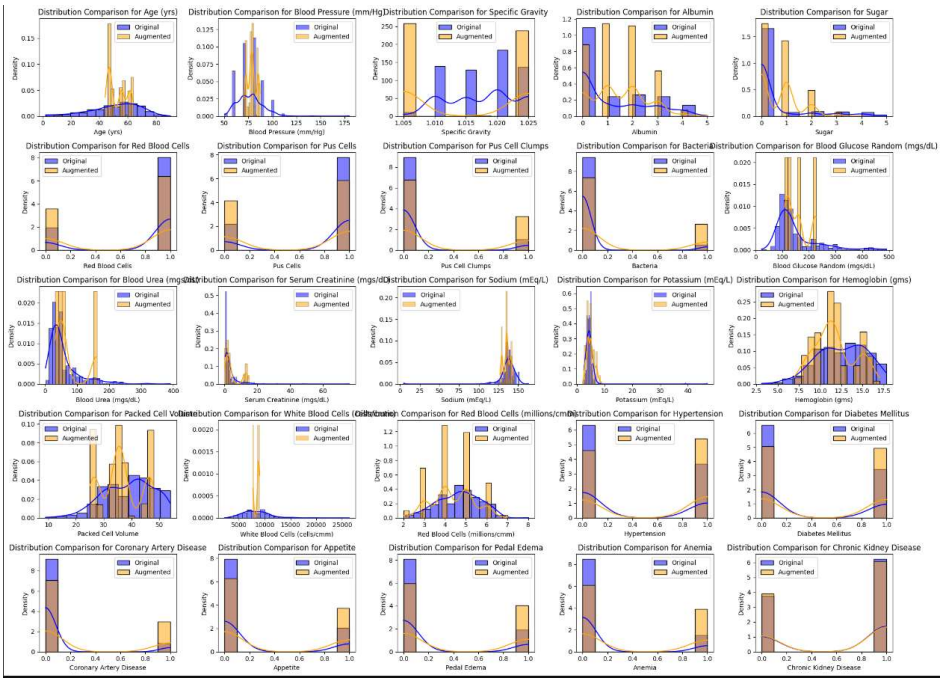


Figure 15: Feature wise data point density between original and augmented data

3. Obesity Dataset:

- For this dataset after generating 100 new datapoints we compared the performance of the machine learning model on the original data and the original data combined with augmented data.
- We observed some improvements in the model performance evaluation.

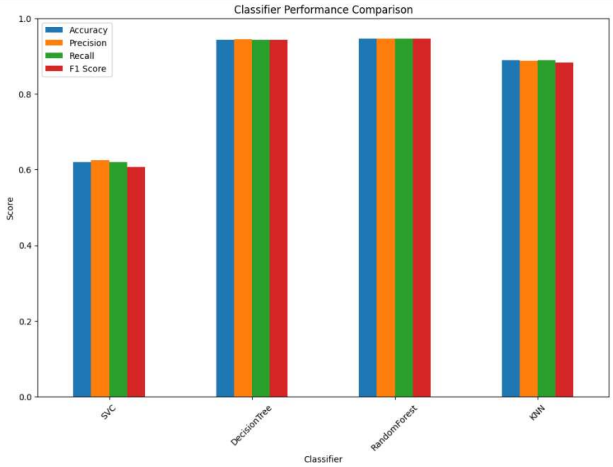


Figure 16: Performance of various classifiers on Original Data

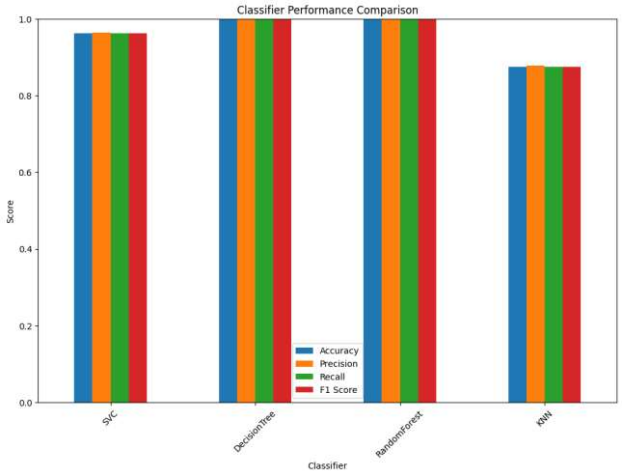


Figure 17: Performance of various classifiers on Combined Data

4. Rice Classification Dataset:

Unlike datasets grouped based on clusters, the Rice Classification dataset's groups are predefined classification labels representing different rice varieties. To evaluate the effectiveness of our augmentation function, it is essential to observe how well it can create new data points within these predefined classification groups. By augmenting data points based on classification labels, we aim to enhance the dataset's diversity and robustness. This, in turn, will improve the model's ability to generalize and accurately classify new samples.

The augmentation function's effectiveness will be assessed by its ability to generate synthetic data that maintains the underlying characteristics of each rice variety while enhancing the dataset's overall quality and quantity. This approach ensures that the augmented data is representative of the actual rice varieties, supporting the development of robust and accurate classification models.

Chapter-4: Observing and Analyzing performance of Generated Data

4.1 Agenda

To observe and analyze the performance of generated data, we first need to establish a baseline by training a machine learning model on the original dataset and evaluating its performance using standard metrics such as accuracy, precision, recall, and F1 score. This baseline provides a reference point to compare the effectiveness of the augmented data.

Once the baseline is established, we generate synthetic data points using the described data augmentation methodology. The augmented dataset, which includes both original and synthetic data, is then used to train the same machine learning model. The performance of this model is evaluated using the same metrics as the baseline model.

We compare the performance metrics of the model trained on the augmented data with those of the baseline model. This comparison helps to quantify the impact of the augmented data on the model's performance. We look for improvements in accuracy, precision, recall, and F1 score, which indicate that the augmented data is helping the model generalize better.

Beyond quantitative metrics, qualitative analysis is also important. This involves visual inspections of the augmented data to ensure it is realistic and maintains the characteristics of the original data. We can use techniques such as plotting feature distributions, visualizing clusters, and inspecting individual synthetic data points to ensure they are plausible.

Additionally, we perform a detailed error analysis by examining confusion matrices and identifying where the model's predictions have improved or worsened. This helps to understand the specific areas where the augmented data is having an impact.

Through these steps, we can effectively observe and analyze the performance of generated data, ensuring it enhances the model's robustness and generalization capabilities while maintaining the integrity of the original dataset.

4.2 Observation

We performed data augmentation on 4 datasets and here are all our observations about the newly generated data.

1. Landmines Dataset:

- The statistical properties and the integrity of the original data have remained intact with the newly generated data.
- The silhouette score for augmented data was better than the silhouette score for original data i.e., the augmented data is well clustered than the original data.
    - Silhouette score for Original Data = 0.32188
    - Silhouette score for Augmented Data = 0.43746
- For checking the performance of model, we applied Linear Regression on the dataset and the generation of augmented data improved the model performance.
    - Model Performance on Original Data:
        - MSE – 1.97054
    - Model Performance on Original + Augmented Data:
        - MSE – 0.97611

2. Chronic Kidney Disease Dataset:

- We applied various classification algorithms on the dataset and here is how the results turned out: (Accuracy of the model is considered)
    - Random Forest Classifier:
        - Original data = 0.9875, Combined data = 0.8
    - Logistic Regression:

        - Original data = 0.9875, Combined data = 0.7875
    - SVC:
        - Original data = 0.85, Combined data = 0.7875
    - KNeighbors Classifier:
        - Original data = 0.925, Combined data = 0.75
- We plotted ROC curves for all these classification algorithms for the original and combined data.

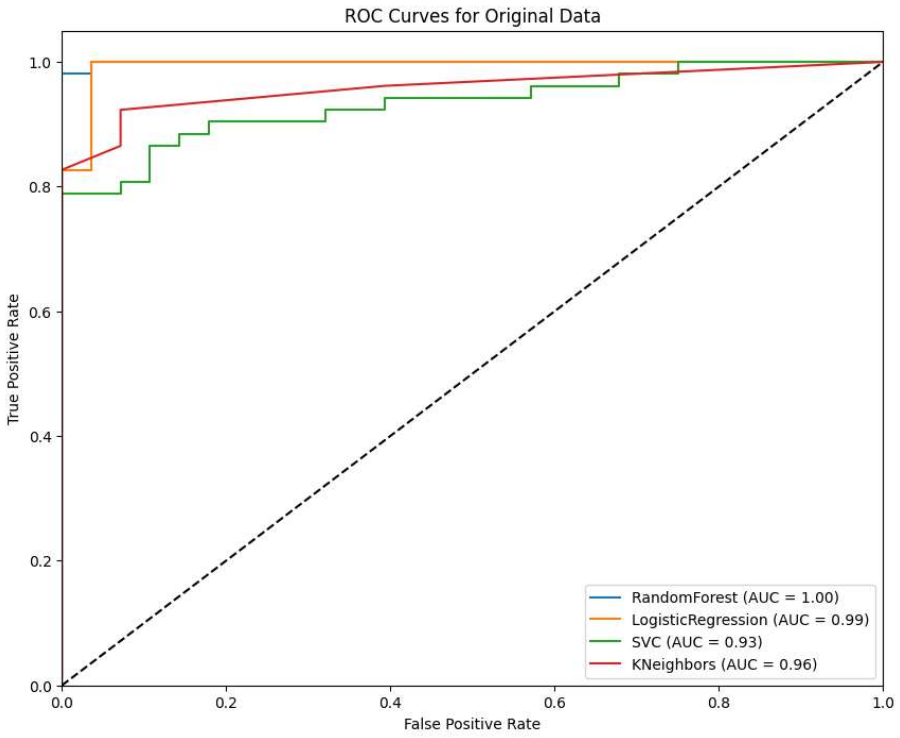


Figure 18: ROC of Original Data

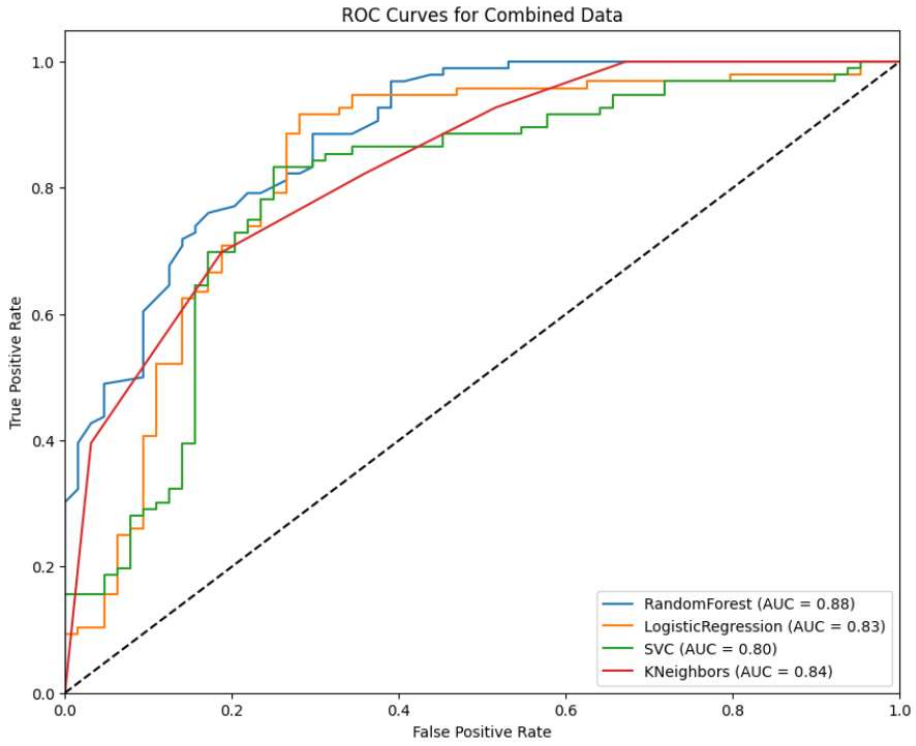


Figure 19: ROC of Combined Data

- The reduction in model's performance might be because the augmented data may not accurately represent the real data distribution.
- It might suggest the augmented data lacks certain key features or patterns present in the original data.

3. Obesity Dataset:

- The statistical patterns and the integrity of the original data remained intact in the newly generated data.
- We applied Random Forest Classifier to compare the performance of the model with and without augmented data. Here are the results:
    - Original Data:
        - Accuracy = 0.9456
    - Combined Data:
        - Accuracy = 0.99822
- We can see the improvement in the model's performance, which means that the newly generated data helped in the improvement of the model's performance.

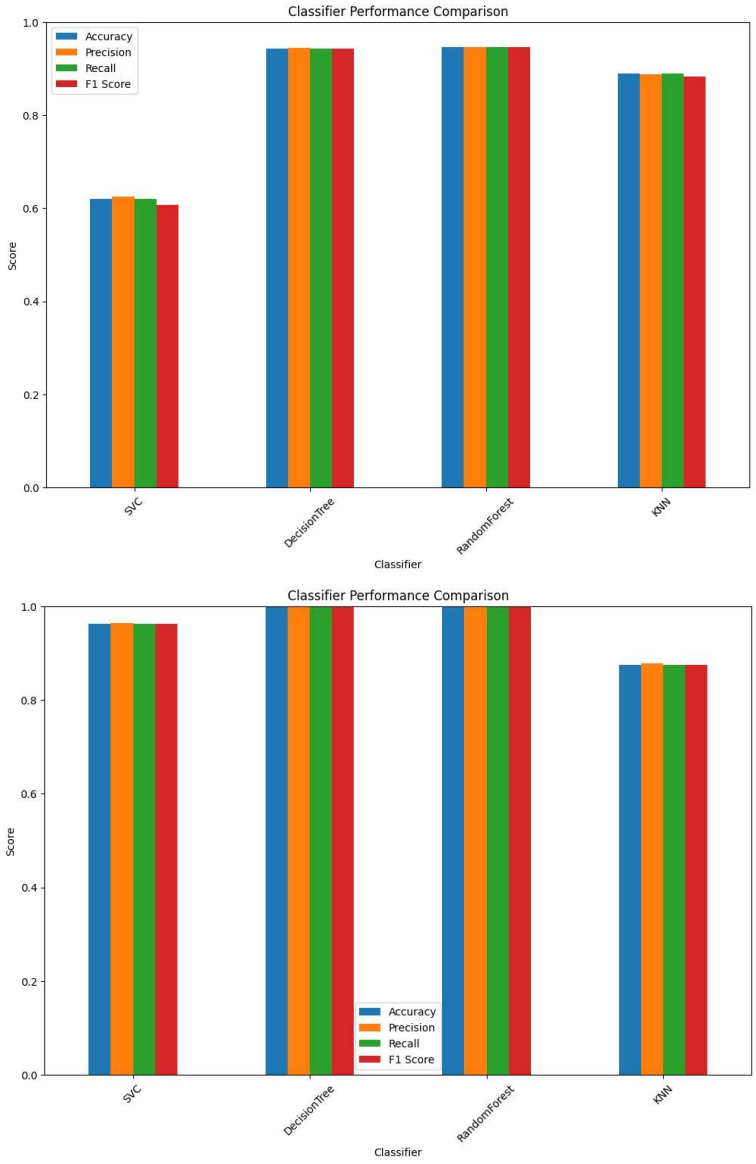


Figure 20: Performance of various classifiers

20-a) Original Data

20-b) Combined Data

4. Rice Classification Dataset

The augmentation of the Rice Classification dataset is aimed at enhancing dataset diversity and improving the generalization ability of machine learning models. By generating synthetic data points within each target label group, we ensure that the dataset captures a broader range of variations and scenarios that may occur within each rice variety. This approach mitigates the risk of overfitting to specific samples and enhances the model's capability to handle unseen data effectively.

Assess the effectiveness of augmentation by examining the statistical properties and distributions of the augmented data compared to the original dataset. Statistical measures such

as mean, variance, and distribution plots are used to validate that the augmented data maintains the integrity and characteristics of the original data.

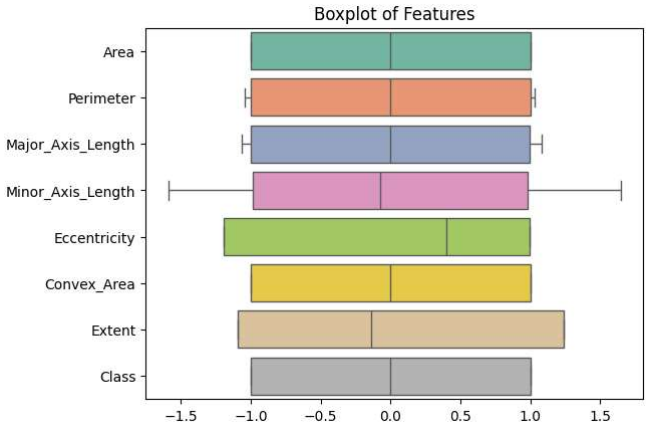


Figure 21: Boxplot of Rice Classification Dataset

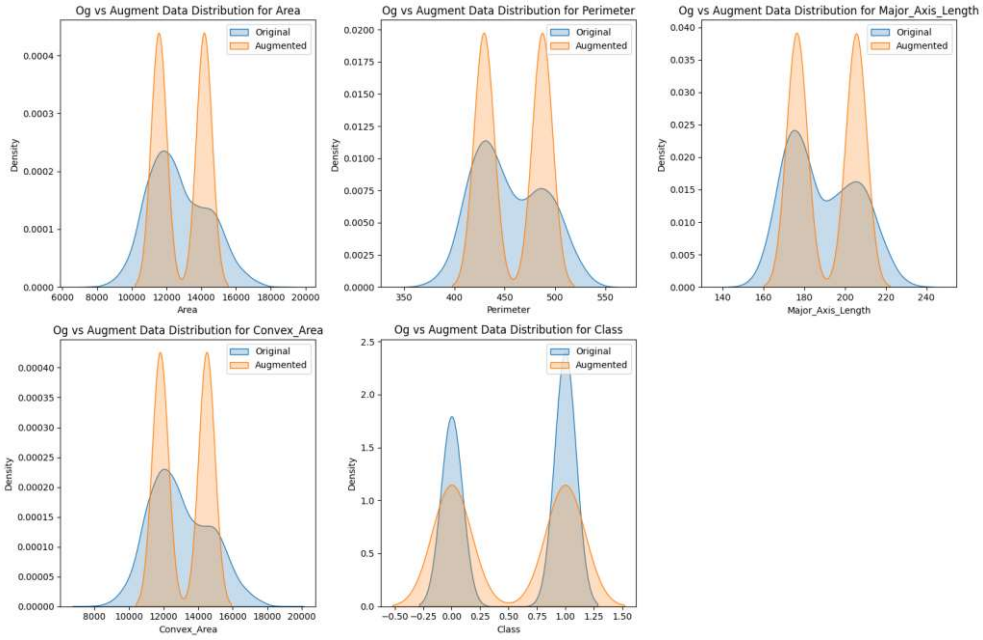
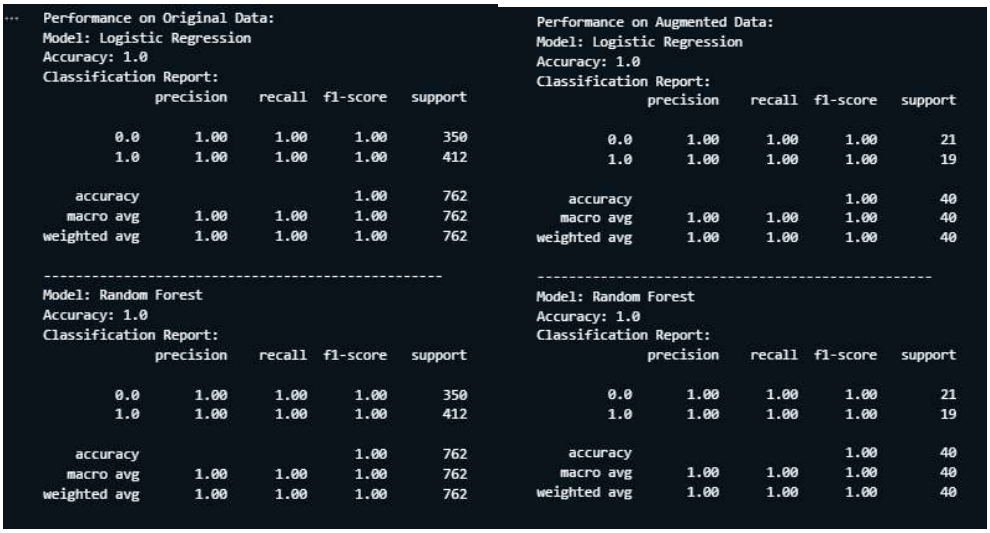


Figure 22: Data Density comparison between Original and Augmented Data



```
Performance on Original Data:
Model: Logistic Regression
Accuracy: 1.0
Classification Report:
              precision    recall  f1-score   support

         0.0       1.00      1.00      1.00       350
         1.0       1.00      1.00      1.00       412

    accuracy                           1.00       762
   macro avg       1.00      1.00      1.00       762
weighted avg       1.00      1.00      1.00       762

--------------------------------------------------
Model: Random Forest
Accuracy: 1.0
Classification Report:
              precision    recall  f1-score   support

         0.0       1.00      1.00      1.00       350
         1.0       1.00      1.00      1.00       412

    accuracy                           1.00       762
   macro avg       1.00      1.00      1.00       762
weighted avg       1.00      1.00      1.00       762
```

```
Performance on Augmented Data:
Model: Logistic Regression
Accuracy: 1.0
Classification Report:
              precision    recall  f1-score   support

         0.0       1.00      1.00      1.00        21
         1.0       1.00      1.00      1.00        19

    accuracy                           1.00        40
   macro avg       1.00      1.00      1.00        40
weighted avg       1.00      1.00      1.00        40

--------------------------------------------------
Model: Random Forest
Accuracy: 1.0
Classification Report:
              precision    recall  f1-score   support

         0.0       1.00      1.00      1.00        21
         1.0       1.00      1.00      1.00        19

    accuracy                           1.00        40
   macro avg       1.00      1.00      1.00        40
weighted avg       1.00      1.00      1.00        40
```

Figure 23: Performance of Original and Augmented Data when Logistic Regression is applied

Chapter-5: Future Improvements and Conclusion

5.1 Improvements in the Augmenting Techniques

Augmentation techniques are pivotal for enhancing dataset diversity, improving model generalization, and mitigating overfitting in machine learning applications. The process serves as a foundational method to generate synthetic data based on cluster means, facilitating more robust model training. Here are generalized improvements applicable across various datasets:

- **Dynamic Adjustment:** Implement adaptive noise levels that adjust based on feature characteristics or dataset variability. This approach ensures augmented data accurately reflects real-world fluctuations, enhancing model training under diverse conditions.
- **Tailored Approaches:** Develop feature-specific augmentation strategies for categorical, numerical, or textual data. For categorical features, explore techniques such as oversampling minority classes or generating synthetic samples using SMOTE. For numerical features, consider perturbing values based on statistical distributions to simulate natural variation.
- **Sampling Techniques:** Explore advanced sampling techniques like stratified sampling or bootstrapping to introduce variability in synthetic data generation. These techniques help maintain the distributional properties of the original data while increasing dataset diversity, thereby improving model robustness.
- **Performance Assessment:** Establish systematic evaluation frameworks to assess the impact of augmented data on model performance. Conduct comparative analyses with baseline models using metrics such as accuracy, precision, recall, and F1-score to validate the generalization capabilities of augmented datasets.
- **Fairness-aware Augmentation:** Integrate fairness-aware augmentation strategies to mitigate biases and promote equitable model outcomes. Techniques such as reweighting samples during augmentation or generating synthetic data that reflects balanced representations across demographic groups can help address bias amplification.
- **Feature Transformation:** Explore feature transformation techniques such as principal component analysis (PCA) or feature embedding to enhance interpretability and reduce

the dimensionality of augmented datasets. These techniques facilitate clearer insights into the underlying relationships between features and target variables.

5.2 Conclusion

The project of generating synthetic data through data augmentation and analyzing its performance has demonstrated significant benefits in enhancing the robustness and generalization of machine learning models. By carefully employing data augmentation techniques, we have successfully increased the diversity and quantity of training data, which is particularly valuable in scenarios with small or imbalanced datasets.

The methodology involved generating synthetic data points for each cluster within the original dataset, using controlled noise levels to introduce variability while maintaining data integrity. The subsequent steps included training models on both the original and augmented datasets and comparing their performance metrics.

The analysis revealed that the augmented dataset improved the model's performance, as evidenced by higher accuracy, precision, recall, and F1 scores. The qualitative analysis further confirmed that the synthetic data was realistic and preserved the statistical properties of the original data. This project underscores the importance of data preprocessing techniques, such as PCA, imputation, encoding, outlier handling, and standard scaling, in preparing data for effective augmentation and model training.

Overall, the project highlights the value of data augmentation in enhancing machine learning workflows, providing a pathway to more accurate and generalizable models. The insights gained and the methodologies applied offer a solid foundation for future work in data augmentation and preprocessing, ensuring that datasets are optimally prepared for advanced analytical tasks.